UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/725,110	12/01/2003	Karlheinz Dorn	P02,0630-01	3328

7590 05/29/2007

SCHIFF HARDIN & WAITE
Patent Department
6600 Sears Tower
233 South Wacker Drive
Chicago, IL 60606

EXAMINER	
LOUIE, OSCAR A	
ART UNIT	PAPER NUMBER
2109	
MAIL DATE	DELIVERY MODE
05/29/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/725,110	DORN ET AL.	
	Examiner	Art Unit	
	Oscar A. Louie	2109	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *18 April 2007*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-11* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-11* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

This final action is in response to the amendment filed on 04/18/2007. Claims 1-11 are pending and have been considered as follows.

Claim Rejections - 35 USC § 101

1. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

- Claim 10 is rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. In claim 10, the applicant is claiming a computer program and what appears to either be another computer program or data which is non-statutory subject matter as in accordance to 35 U.S.C. 101.

Claim Rejections - 35 USC § 102

2. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

3. Claims 1-5, 7-11 are rejected under 35 U.S.C. 102(b) as being anticipated by Dutcher (US-6021496-A).

Claim 1:

Dutcher discloses a method for logging a new user into a data processing device with an operating system and an accessible element that is at least one of an application program and sensitive data, comprising,

- "endinq a first user's access to the accessible element without unloading or restarting the accessible element" (i.e. "the maintenance of user accounts occurs as part of the user logging off form the Windows NT client which is a positive outcome at step 106 of FIG. 11. At logoff, the user account is checked to determine if it is to be managed by the invention. This is determined by checking to see if the user is part of the Roaming Users group on the Windows NT client") [column 12 lines 8-14];
- "determining authentication data for authenticating a user" (i.e. "In this known technique, the gina module 15 tightly controls the locations that are available for authentication to include the local NT workstation itself, the remote NT server 12a, and any other servers that are "trusted" by the NT server that the client is configured against. Generally, only these options are shown to the user seeking authentication, and there are no interfaces available to enable the user to be authenticated from non-native server domains. The present invention addresses this problem") [column 5 lines 22-30];
- "defining an identity and access rights depending on the authentication data" (i.e. "Thus, according to a primary goal of the present invention, the homogeneous NT client-server environment is uncoupled so that a user of a Windows NT client (by way of example only) may be authenticated by a non-native server. With respect to authentication of the Windows NT client, the client-server environment is "heterogeneous." Authentication at

the client gives the user access to resources on the client system, and when this is done via an account definition held at a server, it also gives the user access to resources at the server network via a single logon. The present invention thus enables a user to select a particular location against which he or she desires to be authenticated. Thus, the user's account information may be retained at the non-native server domain in addition to (or instead of) the Windows NT server normally coupled to the Windows NT client in a closed manner. The user's single userid and password are then held out at a non-native server, such as a Warp Server, a DCE cell, or the like. This information may also be retained at a native server domain") [column 6 lines 1-18];

- "providing access, depending on the defined access rights, for at least one of the application program and sensitive data" (i.e. "Thus, according to a primary goal of the present invention, the homogeneous NT client-server environment is uncoupled so that a user of a Windows NT client (by way of example only) may be authenticated by a non-native server. With respect to authentication of the Windows NT client, the client-server environment is "heterogeneous." Authentication at the client gives the user access to resources on the client system, and when this is done via an account definition held at a server, it also gives the user access to resources at the server network via a single logon. The present invention thus enables a user to select a particular location against which he or she desires to be authenticated. Thus, the user's account information may be retained at the non-native server domain in addition to (or instead of) the Windows NT server normally coupled to the Windows NT client in a closed manner. The user's single userid and password are then held out at a non-native server, such as a Warp Server, a DCE cell,

or the like. This information may also be retained at a native server domain") [column 6 lines 1-18];

- "the method being independent of restarting the operating system or the application program" (i.e. "Referring now to FIG. 12, a block diagram is shown of the preferred architecture of the present invention. gina module 15' (ibmgina.dll) exports a set of functions 120 (also referred to as WIx* functions) required to support the WinLogon process. This module also controls the visual elements of the interface including displaying the logon panel, collecting the userid and password from the user, displaying messages, etc. To actually perform the work of authentication, the gina module 15' issues calls to the domain manager 122, which is implemented by dm.dll 124. The domain manager 122 provides the framework that support multiple authentication providers (domain drivers) at the same time. It accepts requests from the gina module ibmgina.dll, determines the appropriate domain driver to handle the request, and then routes the request to the domain driver to actually perform the work. The domain manager 122 also manages dynamically-created local accounts when performing a non-native logon so that the user has the proper security context on his or her workstation when logged on to the server. This frees the domain drivers from re-implementing the same function so that they can concentrate on providing code that is unique to the driver") [column 14 lines 17-38].

Claim 2:

Dutcher discloses a method for logging a new user into a data processing device as in Claim 1 above, and further discloses,

- "displaying a user interface, depending on the defined access rights" (i.e. "Following selection of a domain, the user is then authenticated at step 38. Such authentication may be at a native or non-native server according to the present invention. Typically, this authentication is a process in which the userid and password are provided to a user account database for validation. Upon successful validation, a positive confirmation is received for the authentication and the user processing is allowed to continue. Thereafter, at step 40, when authentication is at a non-native server, a "local" user account is created dynamically (or, alternatively, updated if the user account already exists) at the client machine. This is a Windows NT account in the preferred embodiment. At step 42, the NT user profile is retrieved and established at the client to enable the user to initialize a personal "desktop" and to implement certain access "preferences" at the client. The "user profile" (which normally differs from the "user account" described above) thus preferably includes, without limitation, a desktop definition and a set of preferences for the user. A user profile is created as the user changes appearance and preferences while using the client. Thus, for example, the display screen format is accessed and altered through known techniques (e.g., the Windows '95 desktop "Preferences")") [column 6 lines 43-65];

- "performing a user switch process step that causes the method to begin again at the first step" (i.e. "By supporting "user profiles," the invention provides desktop and environment consistency. Instead of having a single user tied to a single workstation with their own preferences, the users can be "roaming users" that utilize any of a set of workstations. This also supports multiple users being able to get their unique desktops on a single NT client. Further, when the user account is established, the user may become a member of groups having access privileges. These privileges are typically set by system policies that control the functions clients are able to execute. With policies, server administrators can centrally control what users can do on a particular set of servers") [column 13 lines 51-62];
- "content of a user interface remaining unchanged until access rights have been defined again" (i.e. "The routine begins at step 106 to determine whether the user account is to be cleaned up. Step 106 has a positive outcome at logoff, but there may be other occasions when the user is still logged on when it will be desirable to implement the routine. If the outcome of the test at step 106 is negative, the routine cycles. Upon a positive outcome, however, the routine executes a test at step 108 to determine whether the user account should be maintained. If so, a second test is performed at step 110 to determine whether the account should be maintained but disabled. If the outcome of the test at step 110 is negative, the account information is retained on the machine as active at step 112. If the result of the test at step 110 is positive, the account is maintained but disabled at step 114. If, however, the outcome of the test at step 108 is negative, the user account is deleted at step 116 and the routine terminates") [column 11 lines 26-41].

Claim 3:

Dutcher discloses a method for logging a new user into a data processing device as in Claim 2 above, and further discloses,

- "content of the user interface is reduced if the renewed definition of access rights defines a more limited scope than the previous definition allowed" (i.e. "The routine begins at step 106 to determine whether the user account is to be cleaned up. Step 106 has a positive outcome at logoff, but there may be other occasions when the user is still logged on when it will be desirable to implement the routine. If the outcome of the test at step 106 is negative, the routine cycles. Upon a positive outcome, however, the routine executes a test at step 108 to determine whether the user account should be maintained. If so, a second test is performed at step 110 to determine whether the account should be maintained but disabled. If the outcome of the test at step 110 is negative, the account information is retained on the machine as active at step 112. If the result of the test at step 110 is positive, the account is maintained but disabled at step 114. If, however, the outcome of the test at step 108 is negative, the user account is deleted at step 116 and the routine terminates") [column 11 lines 26-41].

Claim 4:

Dutcher discloses a method for logging a new user into a data processing device as in Claim 3 above, and further discloses,

- "generating warning message indicating a reduction in content and that the user has an opportunity to begin the method at the first step again before reduction" (i.e. "WIxDisplayLockedNotice ()--displays the "locked workstation" notice") [column 15 lines 27-28].

Claim 5:

Dutcher discloses a method for logging a new user into a data processing device as in Claim 1 above, and further discloses,

- "displaying a user interface in accordance with the access rights that are defined" (i.e. "Following selection of a domain, the user is then authenticated at step 38. Such authentication may be at a native or non-native server according to the present invention. Typically, this authentication is a process in which the userid and password are provided to a user account database for validation. Upon successful validation, a positive confirmation is received for the authentication and the user processing is allowed to continue. Thereafter, at step 40, when authentication is at a non-native server, a "local" user account is created dynamically (or, alternatively, updated if the user account already exists) at the client machine. This is a Windows NT account in the preferred embodiment. At step 42, the NT user profile is retrieved and established at the client to enable the user to initialize a personal "desktop" and to implement certain access "preferences" at the client. The "user profile" (which normally differs from the "user account" described above) thus preferably includes, without limitation, a desktop definition and a set of preferences for the user. A user profile is created as the user changes appearance and preferences while using the client. Thus, for example, the display screen format is

accessed and altered through known techniques (e.g., the Windows '95 desktop "Preferences")") [column 6 lines 43-65];

- "deleting, by a User Logout procedure, content of a user interface" (i.e. "The routine begins at step 106 to determine whether the user account is to be cleaned up. Step 106 has a positive outcome at logoff, but there may be other occasions when the user is still logged on when it will be desirable to implement the routine. If the outcome of the test at step 106 is negative, the routine cycles. Upon a positive outcome, however, the routine executes a test at step 108 to determine whether the user account should be maintained. If so, a second test is performed at step 110 to determine whether the account should be maintained but disabled. If the outcome of the test at step 110 is negative, the account information is retained on the machine as active at step 112. If the result of the test at step 110 is positive, the account is maintained but disabled at step 114. If, however, the outcome of the test at step 108 is negative, the user account is deleted at step 116 and the routine terminates") [column 11 lines 26-41];
- "starting the method from the first step again" (i.e. "Turning now to FIG. 8, a flowchart is shown of the next step according to the present invention, namely, the establishment of a user account at the client. This was step 40 in FIG. 4. The user account is dynamically established at the client machine in a format of the native operating system. Thus, in the preferred embodiment, a Windows NT user account is established at the client machine after authentication (which may be, as noted above, from a non-native server domain). The routine to dynamically create an NT user begins at step 84 to test whether notification of a successful authentication has been received from the server. If the

outcome of the test at step 84 is negative, the routine cycles. If, however, the outcome of the test at step 84 is positive, the routine continues to create a new NT user on that machine (or update an existing account) at step 85 and to associate a set of access rights to the new (or updated) user account. To this end, the routine continues at step 86 by issuing a request to the server (at which the client was authenticated) to retrieve unique user information and, further, to identify each group in which the user is a member. Although not meant to be limiting, a particular "group" is merely a collection of users that have defined access rights according to some policy. The group information is a convenient mechanism to define the user's privileges with respect to information available from the server. At step 88, the unique user information and group information associated with the authenticated user is retrieved from the server. At step 90, a representation of the "groups" is set up at the client machine. These local "groups" mirror their counterparts on the server. The routine then continues at step 92 to make the user a member of the local groups. This is achieved by linking the user account to the local group information in a data structure. This completes the processing") [column 9 lines 39-67 & column 10 lines 1-4].

Claim 7:

Dutcher discloses a method for logging a new user into a data processing device as in Claim 1 above, and further discloses,

- "activating a screen saver by a defined condition to make a user interface illegible" (i.e. "WIxscreenSaverNotify ()--handles screen saver display request") [column 15 lines 33-34];
- "beginning the method from the first step again" (i.e. "WIxscreenSaverNotify ()--handles screen saver display request") [column 15 lines 33-34].

Claim 8:

Dutcher discloses a method for logging a new user into a data processing device as in Claim 7 above, and further discloses,

- "defined condition is some amount of elapsed time" (i.e. "WIxscreenSaverNotify ()--handles screen saver display request") [column 15 lines 33-34].

Claim 9:

Dutcher discloses a method for logging a new user into a data processing device as in Claim 1 above, and further discloses,

- "blocking all access rights based upon a failed attempt to authenticate a user in the first step" (i.e. "Turning now to FIG. 8, a flowchart is shown of the next step according to the present invention, namely, the establishment of a user account at the client. This was step 40 in FIG. 4. The user account is dynamically established at the client machine in a format of the native operating system. Thus, in the preferred embodiment, a Windows NT user account is established at the client machine after authentication (which may be, as noted above, from a non-native server domain). The routine to dynamically create an NT user begins at step 84 to test whether notification of a successful authentication has been received from the server. If the outcome of the test at step 84 is negative, the routine

cycles. If, however, the outcome of the test at step 84 is positive, the routine continues to create a new NT user on that machine (or update an existing account) at step 85 and to associate a set of access rights to the new (or updated) user account. To this end, the routine continues at step 86 by issuing a request to the server (at which the client was authenticated) to retrieve unique user information and, further, to identify each group in which the user is a member. Although not meant to be limiting, a particular "group" is merely a collection of users that have defined access rights according to some policy. The group information is a convenient mechanism to define the user's privileges with respect to information available from the server. At step 88, the unique user information and group information associated with the authenticated user is retrieved from the server. At step 90, a representation of the "groups" is set up at the client machine. These local "groups" mirror their counterparts on the server. The routine then continues at step 92 to make the user a member of the local groups. This is achieved by linking the user account to the local group information in a data structure. This completes the processing") [column 9 lines 39-67 & column 10 lines 1-4].

Claim 10:

Dutcher discloses a computer system, comprising,

- "an accessible element that is at least one of an application program and sensitive data that is accessible by a first user and a subsequent second user without unloading or restarting the accessible element" (i.e. "One of the preferred implementations of the invention is a client application, namely, a set of instructions (program code) in a code module which may, for example, be resident in the random access memory of the

computer. Until required by the computer, the set of instructions may be stored in another computer memory, for example, in a hard disk drive, or in a removable memory such as an optical disk (for eventual use in a CD ROM) or floppy disk (for eventual use in a floppy disk drive), or downloaded via the Internet or other computer network. Thus, the present invention may be implemented as a computer program product for use in a computer") [column 12 lines 65-67 & column 13 lines 1-15];

- "a program stored in a memory element of the computer comprising a software module or algorithm for determining authentication data for authenticating the second user with respect to the accessible element, a software module or algorithm for defining an identity and access rights depending on the authentication data, and a software module or algorithm for providing access, depending on the defined access rights, for the accessible element" (i.e. "an accessible element that is at least one of an application program and sensitive data that is accessible by a first user and a subsequent second user without unloading or restarting the accessible element" (i.e. "One of the preferred implementations of the invention is a client application, namely, a set of instructions (program code) in a code module which may, for example, be resident in the random access memory of the computer. Until required by the computer, the set of instructions may be stored in another computer memory, for example, in a hard disk drive, or in a removable memory such as an optical disk (for eventual use in a CD ROM) or floppy

disk (for eventual use in a floppy disk drive), or downloaded via the Internet or other computer network. Thus, the present invention may be implemented as a computer program product for use in a computer") [column 12 lines 65-67 & column 13 lines 1-15].

Claim 11:

Dutcher discloses a computer readable data storage media having a program, comprising,

- "a software module or algorithm for determining authentication data for authenticating a user into a data processing device with an operating system and an accessible element that is at least one of an application program and sensitive data" (i.e. "In the prior art, a user of a Windows NT client 10a is typically authenticated by a Windows NT server 12a as shown in FIG. 2. This proprietary client-server linkage is created by the fact that both the client and server run the same operating system (e.g., Microsoft Windows NT 4.0) and use an undocumented set of interfaces for the function of user authentication") [column 4 lines 66-67 & column 5 lines 1-5];
- "a software module or algorithm for defining an identity and access rights depending on the authentication data" (i.e. "In the prior art, a user of a Windows NT client 10a is typically authenticated by a Windows NT server 12a as shown in FIG. 2. This proprietary client-server linkage is created by the fact that both the client and server run the same operating system (e.g., Microsoft Windows NT 4.0) and use an undocumented set of interfaces for the function of user authentication") [column 4 lines 66-67 & column 5 lines 1-5];

- "a software module or algorithm for providing access by the user, depending on the defined access rights, for the accessible element subsequent to an access of the accessible element by a prior first user without unloading or restarting the accessible element" (i.e. "In the prior art, a user of a Windows NT client 10a is typically authenticated by a Windows NT server 12a as shown in FIG. 2. This proprietary client-server linkage is created by the fact that both the client and server run the same operating system (e.g., Microsoft Windows NT 4.0) and use an undocumented set of interfaces for the function of user authentication") [column 4 lines 66-67 & column 5 lines 1-5].

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. Claim 6 is rejected under 35 U.S.C. 103(a) as being unpatentable over Dutcher (US-6021496-A) and in further view of Win (US-6161139-A).

Claim 6:

Dutcher discloses the method as in claim 1 above, but does not disclose,

- "logging all access to the application program and all access to the sensitive data together with the respectively defined identity"

however, Win does disclose,

- "For each login attempt, the Login Tracking Service logs the user's login activity. It saves the time of last successful and unsuccessful logins and number of consecutive, unsuccessful login attempts. The last successful and unsuccessful login times are displayed to the user after each successful login. Users can thus detect if someone else has attempted to use their account" [column 9 lines 46-52];

Therefore, it would have been obvious to one having ordinary skill in the art at the time of the applicant's invention to have logging applied to Dutcher's invention for the purposes of tracking various aspects for security, debugging, and/or troubleshooting since the invention as disclosed by Win entails a network with user logins, where a log or tracker service keeps record of the user activity as is being claimed by the applicant.

Response to Arguments

6. Applicant's arguments filed 04/18/2007 have been fully considered but they are not persuasive. Applicant's arguments regarding independent Claims 1-5 & 7-9 have been considered above in the standing 35 U.S.C. 102(b) rejections and are non-persuasive. It is noted that any computer system comprising user logon and authentication on a network comprises at least two or more users, hence the definition of a computer network. Applicant's arguments regarding Claim 6 have been considered above in the standing 35 U.S.C 103(a) rejection and is

non-persuasive. Win discloses a "Login Tracking Service" that "logs the user's login activity," in a computer network environment with users. This is equivalent to the applicant's "logging," which also operates in a similar environment and logs similar information pertaining to the user's activity.

7. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Conclusion

8. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Examiner Oscar Louie whose telephone number is 571-270-1684. The examiner can normally be reached Monday through Thursday from 7:30 AM to 4:00 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, James Myhre, can be reached at 571-270-1065. The fax phone number for Formal or Official faxes to Technology Center 2100 is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

OAL
05/24/2007

James Myhre
Supervisory Patent Examiner